RESEARCH ARTICLE

Efficiency Optimization Design That Considers Control of Interior Permanent Magnet Synchronous Motors Based on Machine Learning for Automotive Application

YUKI SHIMIZU, (Member, IEEE)

College of Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 5258577, Japan

e-mail: yshimizu@fc.ritsumei.ac.jp

ABSTRACT Interior permanent magnet synchronous motors have become widely used as traction motors in environmentally friendly vehicles. Interior permanent magnet synchronous motors have a high degree of design freedom and time-consuming finite element analysis is required for their characteristics analysis, which results in a long design period. Here, we propose a method for fast efficiency maximization design that uses a machine-learning-based surrogate model. The surrogate model predicts motor parameters and iron loss with the same accuracy as that of finite element analysis but in a much shorter time. Furthermore, using the current and speed conditions in addition to geometry information as input to the surrogate model enables design optimization that considers motor control. The proposed method completed multi-objective multi-constraint optimization for multi-dimensional geometric parameters, which is prohibitively time-consuming using finite element analysis, in a few hours. The proposed shapes reduced losses under a vehicle test cycle compared with the initial shape. The proposed method was applied to motors with three rotor topologies to verify its generality.

INDEX TERMS Finite element analysis, iron loss, machine learning, motor efficiency, multi-objective multi-constraint optimization, permanent magnet motors, XGBoost.

I. INTRODUCTION

Environmentally friendly vehicles have increasingly attracted attention as a tool for achieving carbon neutrality. Interior permanent magnet synchronous motors (IPMSMs) [1], [2], [3] are used as traction motors in such vehicles because of their high output, high efficiency, and high reliability.

IPMSMs have permanent magnets (PMs) inside the rotor. There is a high degree of freedom in the PM arrangement and the flux barrier arrangement, and thus various designs are possible for a given set of specifications. Finite element analysis (FEA) is generally used to analyze the characteristics of IPMSMs; however, this analysis is very time-consuming. For IPMSM analysis for automotive applications, the analysis time is very long because the characteristics must be obtained under various speed and torque conditions (e.g., city driving, suburban driving, highway driving). The design of IPMSMs takes a long time because FEA must be repeatedly performed on various design alternatives. There is thus a need for an automatic design system that can efficiently design high-performance IPMSMs.

Many studies have proposed algorithms for efficiently finding the optimal design. Bonthu et al. [4] minimized torque ripple and cogging torque by optimizing the notch shape of the rotor surface of a PM-assisted synchronous reluctance motor. Farhadian et al. [5] proposed an optimization algorithm based on improved particle swarm optimization to optimize the torque characteristics of a synchronous reluctance motor. Son et al. [6] optimized the rotor shape of an IPMSM with grain-oriented electrical steel applied to the

stator teeth using a modified genetic algorithm. Shi et al. [7] proposed multi-objective optimization based on the sequential Taguchi method for a five-phase permanent magnet hub motor. Although these methods efficiently optimize geometry, they are unsuitable for IPMSMs in automotive applications because they obtain the motor characteristics only at a single current condition.

To speed up the design process, many studies have constructed surrogate models for FEA. A surrogate model functionalizes the relationship between the input (geometry and operating conditions) and the output (the characteristics) of FEA. Using a surrogate model allows us to perform IPMSM design with a reduced number of FEA iterations or without FEA. Building a surrogate model requires some FEA results. Many studies have utilized surrogate models trained by machine learning. Islam et al. [8] optimized two rotor design parameters at multiple output points of an IPMSM using the response surface method. Zheng et al. [9] performed the multi-objective optimization of an IPMSM with rare-earth PMs and ferrite PMs using the response surface method. Sun et al. [10] classified the geometric parameters of an IPMSM into three categories by cross-factor variance analysis and optimized them with respect to torque and loss characteristics using kriging. Sun et al. [11], [12] proposed a system-level design optimization method based on the actual operating environment and a sequential subspace optimization method using the kriging method for a permanent magnet hub motor, respectively. Dhulipati et al. [13] used support vector regression to train a prediction model for a six-phase IPMSM. Hao et al. [14] trained a model to learn the relationship between the design parameters and torque ripple of an IPMSM using radial basis function networks and used the model for optimization. Pan et al. [15] used XGBoost, an optimized distributed gradient boosting library, to learn the relationship between the torque characteristics and the structural parameters of PM arc motors and used the model for optimization. These machine-learning-based surrogate models are very effective for the automated design of IPMSMs.

However, much of the above-mentioned research on design optimization of electromagnetic machines with machine learning did not focus on control strategies. For traction motors for automotive applications, some operating points (torques and speeds) are given. At that operating points, it is necessary to consider designs that optimize the characteristics such as efficiency. Because various current conditions are possible to drive at given operating points, it is necessary to determine the optimal current conditions based on a control strategy, which requires repeated FEA. Especially during design optimization, for each of a large number of design alternatives, the optimal current condition needs to be searched based on a control strategy and requires a high computational cost. Therefore, this study proposes a fast efficiency optimization method for IPMSM for automotive applications with machine-learning-based surrogate models. The contributions of this study are as follows.

- Machine-learning-based surrogate models are proposed for predicting iron loss characteristics based on current and speed conditions in addition to geometry information, which can reduce characteristic computation time compared to FEA.
- A surrogate model is used to obtain the current conditions for driving at the required operating points with copper loss minimization control and to find the best design that maximizes the efficiency characteristics under such control.

The rest of this paper is organized as follows. Section II describes the mathematical modeling of the target IPMSM. Section III describes the training of the surrogate model with machine learning. Section IV shows the results of efficiency maximization with a torque constraint using the proposed surrogate model. Section V summarizes the results.

The dataset described in Section III-C is available at IEEE DataPort [16] and the Python implementation of the iron loss prediction model is available at GitHub [17].

II. MATHEMATICAL MODELING
A. TARGET IPMSM
This study focuses on IPMSMs for automotive applications. Fig. 1 shows the structure of a double-layered IPMSM (Type 2D) and Table 1 shows the corresponding specifications. To verify the generality of the proposed method, this study uses three rotor topologies (see Fig. 4 below). All IPMSMs have 8-pole, 48-slot stators with distributed windings. The details of each model are described elsewhere [3].

B. IPMSM MODELING
The average torque T of an IPMSM is obtained as follows.

$$\begin{aligned} T &= P_n \left\{ \Psi_a i_q + \left(L_d - L_q \right) i_d i_q \right\} \\ &= P_n \left\{ \Psi_a I_a \cos \beta + \frac{1}{2} \left(L_q - L_d \right) I_a^2 \sin 2\beta \right\}, \end{aligned} \quad (1)$$

where P_n is the number of pole pairs, Ψ_a is the PM flux linkage, L_d and L_q are the d- and q-axis inductances, respectively, i_d and i_q are the d- and q-axis currents, respectively, and I_a and β are the magnitude and leading angle with respect to the q-axis of the armature current vector, respectively.

Because IPMSMs are driven under inverter voltage and current limits, the current vector that maximizes the output should be searched for under each limit. The current and voltage limits are respectively expressed by the following equations.

$$I_a = \sqrt{i_d^2 + i_q^2} \leq I_{am}, \quad (2)$$

$$V_a = \sqrt{v_d^2 + v_q^2} \leq V_{am}, \quad (3)$$

where I_{am} is the maximum armature current, V_a is the terminal voltage, v_d and v_q are the d- and q-axis voltages, respectively, and V_{am} is the maximum terminal voltage, which depends on the dc-link voltage. The limit of the induced
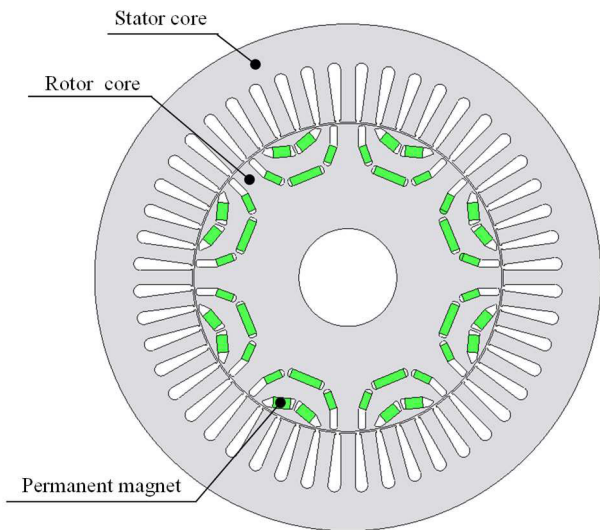
FIGURE 1. Cross section of target traction motors. Rotor topology is Type 2D [3]. Other rotor topologies used in this study are shown in Fig. 4.

TABLE 1. General motor specifications.

Item (Unit)	Value
Number of pole/slot	8/48
Stator diameter (mm)	264
Rotor diameter (mm)	160.4
Air gap length (mm)	0.75
Shaft diameter (mm)	51
Stack length (mm)	50
Winding resistance (Ω)	0.129 (at 180 °C)
Maximum phase current (Arms)	134
Maximum terminal voltage (V)	507
Maximum speed (min^{-1})	14000
Iron core material	10JNEX900
PM material	NMX-S49CH

voltage V_o of the IPMSM is expressed as follows.

$$V_o = \omega\sqrt{\Psi_d^2 + \Psi_q^2}$$
$$= \omega\sqrt{(\Psi_a - L_d I_a \sin\beta)^2 + (L_q I_a \cos\beta)^2} \leq V_{om}, \quad (4)$$

where Ψ_d and Ψ_q are the d- and q-axis flux linkages, respectively, and the maximum induced voltage V_{om} depends on V_{am} as follows.

$$V_{om} = \sqrt{V_{am}^2 + (R_a I_a)^2 - 2V_{am}R_a I_a \cos\phi} \geq V_{am} - R_a I_{am}, \quad (5)$$

where R_a is the winding resistance, and φ is the power factor angle. The equality condition for (5) is the power factor $\cos\varphi = 1$. For simplicity, this study sets the maximum induced voltage V_{om} to the minimum value $V_{am} - R_a I_{am}$.

C. LOSS EVALUATION

This section describes the losses of the IPMSM. IPMSMs for automotive applications are required to perform under various speed and torque conditions. The Worldwide Harmonized Light Vehicles Test Cycle (WLTC) is commonly used
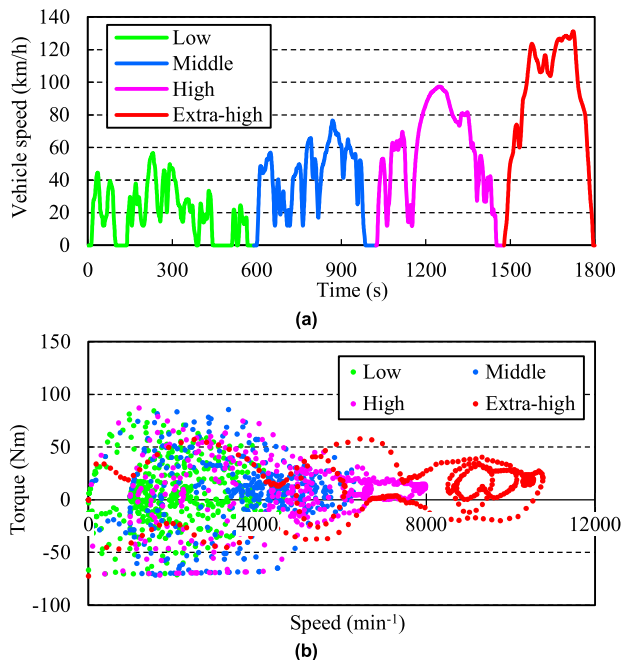

FIGURE 2. WLTC results. (a) Vehicle speed and (b) discrete operating points of motor.

for vehicle loss and efficiency evaluation. Fig. 2 shows the vehicle speed and the discrete operating points of the motor under the WLTC. For the conversion from vehicle speed to motor operating point, the body constants of Toyota's third-generation Prius were used. For more details, see [3].

As shown in Fig. 2, the operating point of a motor is given as a speed and a torque. However, there are various current conditions where the motor can be driven at a single operating point. Therefore, the evaluation of driving characteristics at a certain operating point requires finding the appropriate current conditions. This study finds the appropriate current conditions $(\hat{I}_a, \hat{\beta})$ for a given operating point (ω_{req}, T_{req}) by minimizing copper loss as follows.

$$\hat{I}_a, \hat{\beta} = \underset{I_a \in (0, I_{am}], \beta \in [0, 90]}{\arg\min} R_a I_a^2,$$
$$s.t.\ T(I_a, \beta) = T_{req},$$
$$V_o(I_a, \beta, \omega_{req}) \leq V_{om}. \quad (6)$$

The reason for using copper loss minimization is that copper loss is easy to calculate and is often used in actual motor drives rather than maximum efficiency control, which takes into account iron loss and other losses.

For motors for automotive applications, it is difficult to solve (6) analytically because the motor parameters have strong nonlinearities due to the effect of magnetic saturation. In general, (6) is solved numerically by repeatedly performing FEA. Fig. 3 shows the algorithm for the current condition search based on copper loss minimization, where $\Delta I_a > 0$ and $\Delta\beta > 0$ are the search step sizes for I_a and β, respectively, and $\varepsilon > 0$ is the threshold value. Because copper loss depends only on the current amplitude, copper loss minimization only
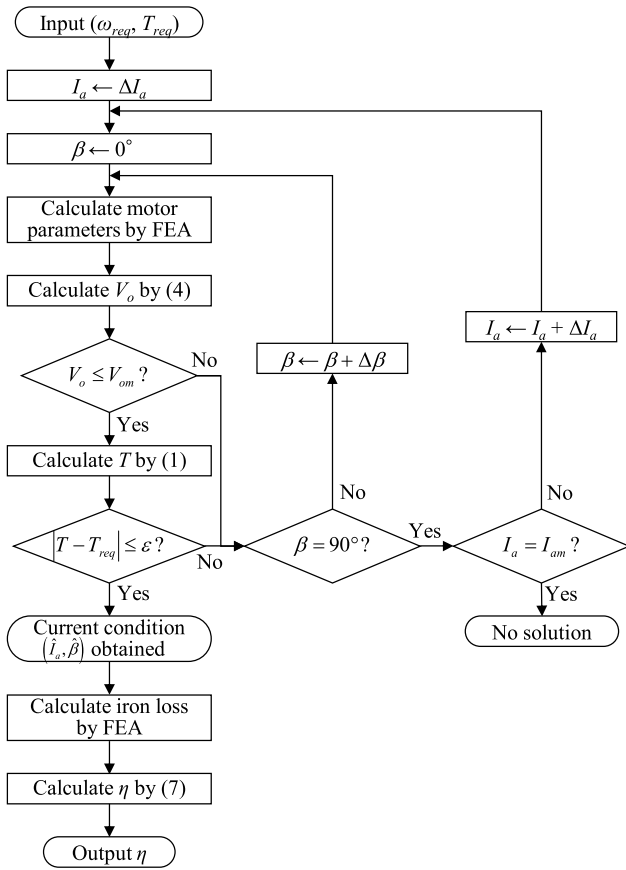
FIGURE 3. Algorithm to find appropriate current conditions by minimizing copper loss and to calculate efficiency.

needs to find the smallest current amplitude condition that satisfies the constraints. Therefore, this algorithm performs the characteristics analysis sequentially starting from the condition with the smallest current amplitude and searches for the smallest current condition that satisfies the constraints.

Once the appropriate current condition is found, the efficiency is calculated. Because this study uses static magnetic field analysis, iron loss does not affect the analyzed power, and iron loss analysis is performed separately under arbitrary speed conditions. Therefore, the efficiency η is calculated as follows.

$$\eta = \frac{\omega_m T - W_i}{\omega_m T + R_a I_a^2}, \tag{7}$$

where ω_m is the mechanical angular velocity, and W_i is the iron loss.

III. MACHINE-LEARNING-BASED SURROGATE MODEL
For the efficiency maximization design, an efficiency evaluation is performed for a large number of solution shapes. However, as described in Section II, the efficiency evaluation of IPMSMs for automotive applications requires repeated FEA, which is time-consuming and thus makes large-scale

optimization design difficult. This study constructs surrogate models that can predict the driving characteristics of an IPMSM from the current and speed conditions and its geometry using machine learning.

A. MOTOR PARAMETER PREDICTION
The parameters to be predicted by machine learning in the efficiency maximization design are motor parameters and the iron loss. The motor parameters include PM flux linkage Ψ_a and d- and q-axis inductances L_d and L_q, respectively. Using (1) and (4) with these parameters, the average torque and induced voltage can be calculated, respectively.

To take into account the effects of magnetic saturation, the surrogate model that predicts these motor parameters takes as input the current condition I_e and β in addition to the geometric parameters. For the analysis of the motor parameters, note that Ψ_a is calculated from the FEA results at $\beta = 0°$ and that Ψ_a is required for calculating L_d. Based on these relationships, we efficiently generate data and use support vector regression and XGBoost to construct surrogate models for motor parameters with high accuracy. See [18] for details. This study uses the surrogate models constructed in [18] for parameter prediction.

B. ROTOR SHAPE GENERATION
Next, the iron loss prediction method is described. Fig. 4 shows the initial rotor topologies and geometric parameters for the three IPMSMs [3], [19], where $\left(r_1^{\cdot}, \theta_1^{\cdot}\right)$ are the polar coordinates, with the center of the shaft as the origin. The geometric parameter vectors for each topology are defined as

$$\mathbf{x}_{geom}^{2D} = \left(r_1^{2D}, \theta_1^{2D}, d_1^{2D}, d_2^{2D}, \ldots, d_9^{2D}\right)^T \in \mathbb{R}^{11},$$

$$\mathbf{x}_{geom}^{V} = \left(r_1^{V}, \theta_1^{V}, d_1^{V}, d_2^{V}, d_3^{V}\right)^T \in \mathbb{R}^{5},$$

$$\mathbf{x}_{geom}^{Nabla} = \left(d_1^{Nabla}, d_2^{Nabla}, \ldots, d_8^{Nabla}\right)^T \in \mathbb{R}^{8}, \tag{8}$$

and all variables not included in $\mathbf{x}_{geom}^{\bullet}$ or dimensions not shown in Fig. 4 are either constant or automatically determined (e.g., the PMs in the second layer in Fig. 4(a) have the same dimensions). The arc shape at the end of the flux barrier has been removed for simplicity. The rotor shape can be obtained by randomly generating $\mathbf{x}_{geom}^{\bullet}$ according to the following equation.

$$x_{geom(j)}^{\bullet} \sim U(x_{lwr(j)}^{\bullet}, x_{upr(j)}^{\bullet})(j = 1, \ldots, 11), \tag{9}$$

where $U(a, b)$ is a random variable with a uniform distribution on an open interval (a, b), $x_{geom(j)}^{\bullet}$ is the j-th geometric parameter, and $x_{lwr(j)}^{\bullet}, x_{upr(j)}^{\bullet}$ are its upper and lower bounds, respectively. The upper and lower limits of the basic geometric parameters $\mathbf{x}_{geom}^{\bullet}$ are set with inter-variable dependencies to prevent the generation of un-designable shapes (e.g., magnets popping out of the rotor area) [18]. Because the upper and lower bounds for each parameter are different from case
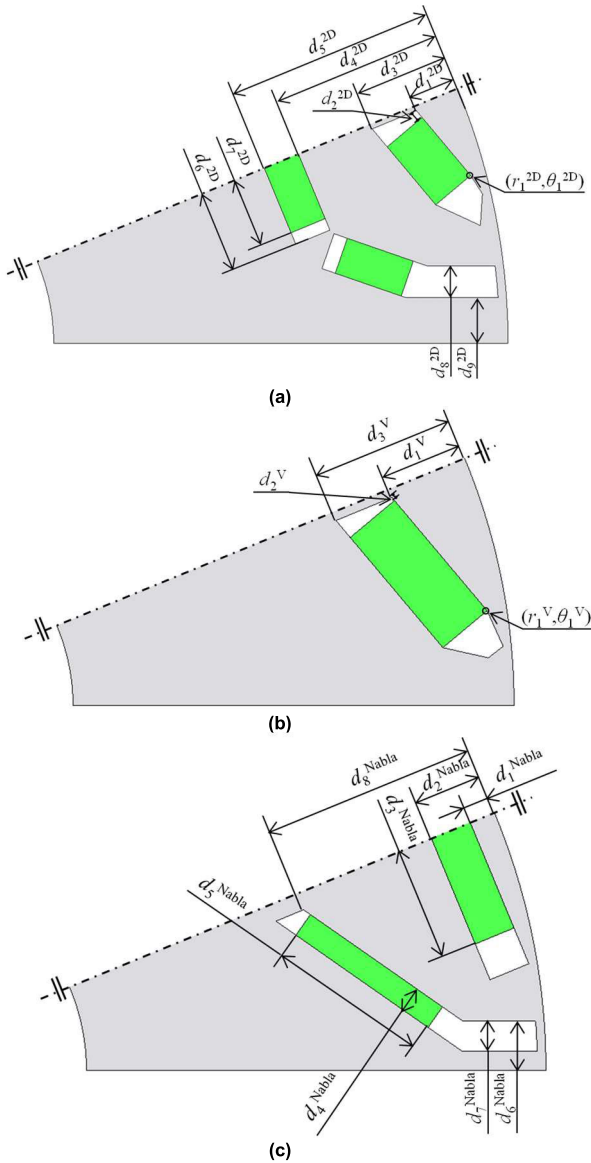
(a)

(b)

(c)

FIGURE 4. Initial rotor shapes and settings for geometric parameters. Dimensions not shown in figure are either constant or automatically determined. (a) 2D [3], (b) V [3], and (c) Nabla [19].

to case, the simple uniform distribution was used instead of complex sampling methods such as the Latin hypercube sampling. The directions of magnetization of the PMs were fixed in the thickness direction.

For 2D, V, and Nabla topologies, 28,990, 28,539, and 27,655 rotor shapes for the training datasets were randomly generated according to (9), respectively. Python was used to implement the uniform distribution. (In fact, 30,000 rotor shapes for each topology were prepared, but excluding the cases where the FEA failed described in Sec. III-C, the numbers of data were as described above.)

C. FINITE ELEMENT ANALYSIS CONDITIONS
This section describes the analysis of the shapes generated in Section III-B. Phase currents I_e, current phase angles β, and

speeds N were generated according to (10).

$$\begin{cases} I_e \sim 140\sqrt{U(0, 1)}(\text{Arms}), \\ \beta \sim U(0, 90)(°), \\ N \sim U(0, 15000) \ (\text{min})^{-1}. \end{cases} \tag{10}$$

This setting means that one geometry is analyzed under only one current and speed condition.

The rotor position was changed in one-degree (mechanical angle) steps in the two-dimensional FEA using JMAG-Designer 19.1 as the analysis software. In the iron loss analysis, the fast Fourier transform method was adopted for both hysteresis loss and eddy current loss computation without considering the minor loop. The eddy current loss in the PM was not considered. For the analysis, a computer with an Intel Core[TM]i9-10980XE CPU and 32.0 GB of RAM was used. The total FEA time for 28,990, 28,539, and 27,655 rotor shapes of the 2D, V, and Nabla topologies was 477.2, 372.5, and 439.8 hours, respectively.

D. XGBOOST
Because the training dataset is tabular and multi-dimensional and contains a lot of data, this study uses XGBoost as the learning method for iron loss prediction. XGBoost is a gradient boosting method proposed by Chen and Guestrin [20]. It uses decision trees as weak learners. The learning process of XGBoost is described below.

First, the first decision tree is adapted from the input data. Next, a second decision tree is fitted to reduce the residuals of the first one. In this way, when K decision trees are constructed, the predicted value $f(\mathbf{x}_i)$ is expressed by (11) with the k-th decision tree as f_k.

$$f(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \tag{11}$$

When constructing the t-th decision tree from t-1 decision trees, the following optimization problem is solved.

$$\min_{f_t} \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)\right) + \Omega(f_t), \tag{12}$$

where $\hat{y}_i^{(t-1)}$ is the prediction by t-1 decision trees, and the loss function l uses the root-mean-square error. Ω is the regularization term defined as follows.

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2, \tag{13}$$

where T is the number of leaf nodes in the decision tree, $w \in R^T$ is the value stored in the leaf node, and $\gamma, \lambda > 0$ are hyperparameters.

E. TRAINING SETTINGS
The input vectors $\mathbf{x}_{ironloss}^{\bullet}$ in iron loss prediction are 14-dimensional vectors.

$$\mathbf{x}_{ironloss}^{\bullet} = \left(i_d, i_q, N, \mathbf{x}_{geom}^{\bullet}\right)^{\text{T}}. \tag{14}$$
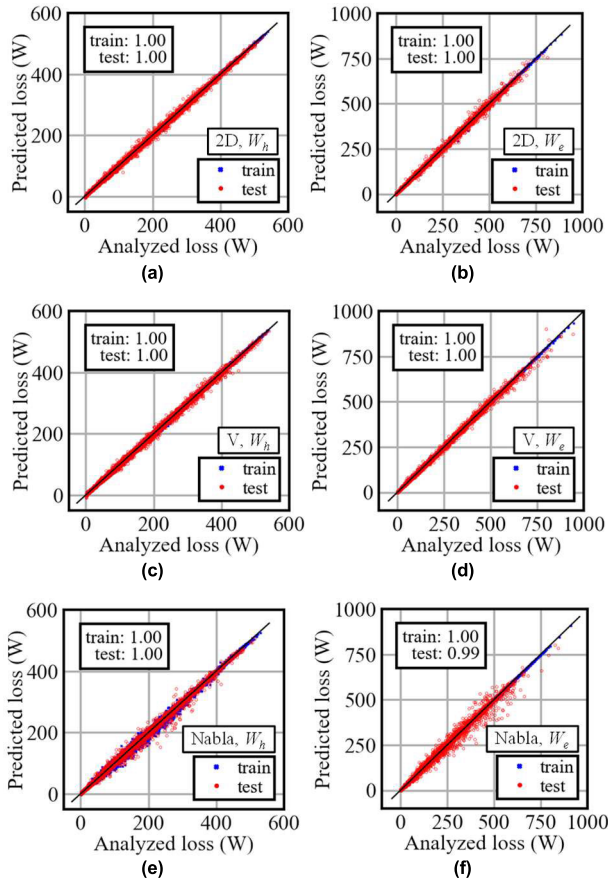
FIGURE 5. Prediction results for iron loss. The upper left number indicates the coefficient of determination (higher is better). (a), (c), and (e) show the results for hysteresis loss, and (b), (d), and (f) show the results for eddy current loss. (a) and (b) are 2D, (c) and (d) are V, and (e) and (f) are Nabla.

As a general formula for iron loss, an empirical equation, called Steinmetz's equation [21], is used.

$$W_i = W_h + W_e,$$
$$W_h = m_{core} k_h f B_{max}^{1.6},$$
$$W_e = m_{core} k_e (f B_{max})^2, \quad (15)$$

where W_h is the hysteresis loss, W_e is the eddy current loss, m_{core} is the core mass, k_h and k_e are constants, f is the frequency, and B_{max} is the maximum flux density. According to (15), the hysteresis loss and the eddy current loss have different nonlinearities with respect to frequency and the maximum magnetic flux density. Therefore, to improve prediction accuracy, the hysteresis loss and the eddy current loss were predicted separately as follows.

$$W_h^{pred} = f_h \left(\mathbf{x}_{ironloss}^{\bullet} \right),$$
$$W_e^{pred} = f_e \left(\mathbf{x}_{ironloss}^{\bullet} \right), \quad (16)$$

The functions f_h and f_e are subject to training with XGBoost.

In the training, the datasets of features and targets were split into two subsets, namely training and test sets, in a 4:1 ratio. The datasets were standardized before training.
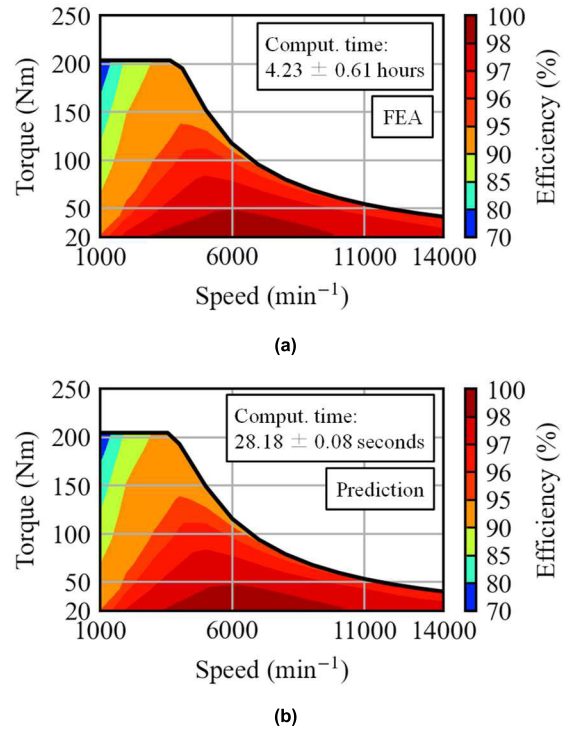

FIGURE 6. Efficiency maps for initial 2D model [3] (a) created by FEA and (b) predicted by machine learning.
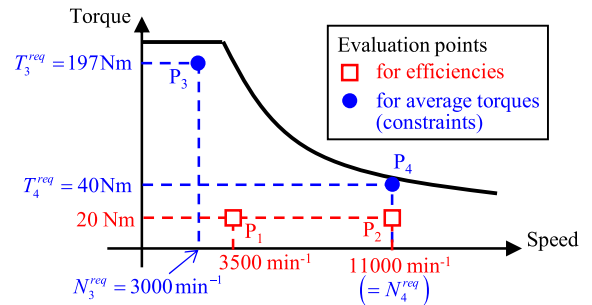

FIGURE 7. Evaluation points used in optimization. The blue points are for efficiency evaluation. The red squares are for average torque constraints.

Python was used for implementation. Five-fold cross-validation with the library Optuna was used for hyperparameter tuning.

F. TRAINING RESULTS

Fig. 5 shows the prediction results of the hysteresis loss and the eddy current loss. The prediction of both hysteresis loss and eddy current loss was found to be highly accurate even for unknown test data. The prediction accuracies were equally high for all topologies.

Fig. 6 compares the efficiency maps of the initial 2D models created by FEA and those predicted by machine learning. The computation time for generating efficiency maps for FEA was 4.23 ± 0.61 hours and that for the prediction by machine learning was 28.18 ± 0.08 seconds. The computation times are the means and standard deviations (std) of 5 trials for
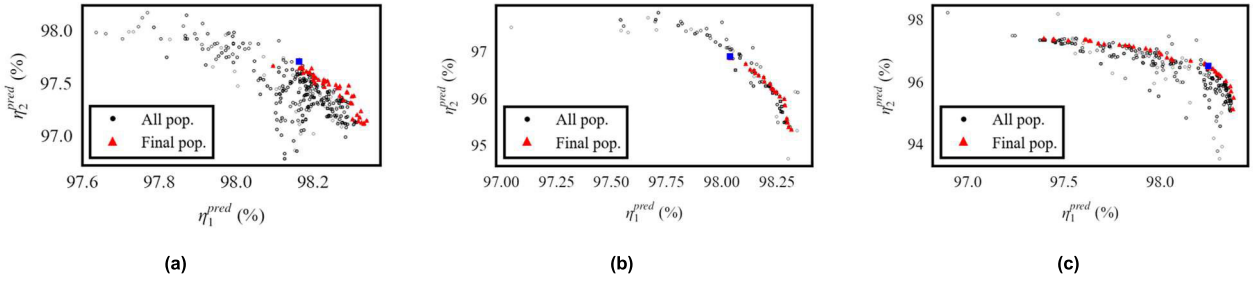
FIGURE 8. Efficiencies for all populations and final population for optimization design. The blue squares are the selected Pareto solution shown in Fig. 10. (a) 2D, (b) V, and (c) Nabla.
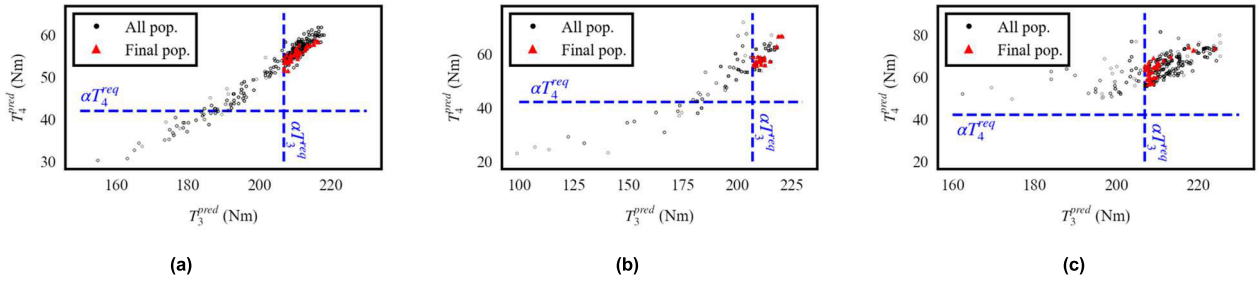

FIGURE 9. Torques for all populations and final population for optimization design. (a) 2D, (b) V, and (c) Nabla.

FEA and 100 trials for prediction. The results of prediction by machine learning are similar to those of FEA, suggesting that this prediction has potential use as a highly accurate surrogate model. The computation time of the surrogate model was less than 0.2% that of FEA, indicating that once training is completed, the surrogate model can be repeatedly used for large-scale design optimization.

IV. EFFICIENCY OPTIMIZATION
In this section, a multi-objective multi-constraint optimization design is performed using the trained prediction models.

A. PROBLEM
It is time-consuming to evaluate efficiency at all operating points of the WLTC. Therefore, the characteristics at two representative points are evaluated. Fig. 7 shows the evaluation points used in optimization. The efficiency evaluation points are determined based on the motor operating points in WLTC as shown in Fig. 2, and two other torque constraints are set to maintain the same output as the reference motor. The efficiency maximization problem with a torque constraint for the IPMSM is as follows.

$$\min_{\mathbf{x}^{\bullet}_{geom}} \quad -w_1 \frac{\eta_1^{pred}}{\eta_1^{init}} - w_2 \frac{\eta_2^{pred}}{\eta_2^{init}} + P_{AD},$$

$$s.t. \; g_i : T_i^{pred} \geq \alpha T_i^{req} \quad (i = 3, 4), \qquad (17)$$

where η_1^{pred} and η_2^{pred} are the predicted efficiencies at operating points P$_1$ and P$_2$, which are calculated by (6), (7), and (16) and normalized by the initial values η_1^{init} and η_2^{init}, respectively. w_1 and w_2 are weight coefficients, $(w_1, w_2) = (1, 1)$ in this setup. P_{AD} is the applicability domain penalty

of the prediction models [18]. The constraint conditions g_i are torque constraints for two required operating points $\{(N_i^{req}, T_i^{req})\}_{i=3}^4$, which are multiplied by a coefficient ($\alpha = 1.05$) to consider the prediction error. The predicted torques used for efficiency calculations and torque constraints were calculated by (1) and the surrogate model for predicting motor parameters described in Section III-A at the current conditions obtained in (6).

The current conditions for the efficiency evaluation were determined by copper loss minimization using the algorithm in Fig. 3 with the surrogate model instead of FEA. This algorithm is implemented using the array computation in the Python library NumPy. The torque prediction results are given as the maximum output control at the required speed using the surrogate models for motor parameters.

NSGA-II [22] was used as the optimization algorithm and the Python library pymoo was used for the implementation [23]. For 2D, V, and Nabla, the population sizes were 50, 25, and 40, and the numbers of offspring were 10, 8, and 10, respectively. These values depended on the dimensions of the geometric parameters. Latin hypercube sampling was used for sampling the initial population, the tournament method was used for selection, simulated binary crossover was used for crossover, and polynomial mutation was used for mutation. The termination condition was set to 50 generations.

B. OPTIMIZATION RESULTS
Figs. 8 and 9 show the efficiency and torque results for all populations and the final population during the optimization process. For all topologies, clear Pareto fronts were generated for η_1^{pred} and η_2^{pred}, which have a trade-off relationship.
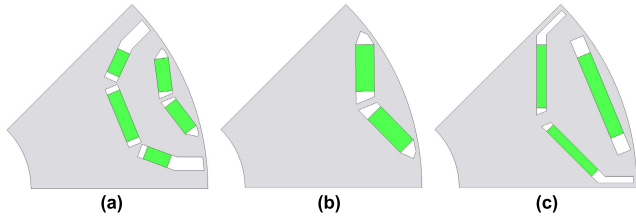
FIGURE 10. Best rotor shapes of selected Pareto solution (blue squares in Fig. 8). (a) 2D, (b) V, and (c) Nabla.

TABLE 2. Geometric parameters for best rotor shapes.

	$r_1^\bullet$ (mm)	$\theta_1^\bullet$ (°)	$d_1^\bullet$ (mm)	$d_2^\bullet$ (mm)	$d_3^\bullet$ (mm)	$d_4^\bullet$ (mm)	$d_5^\bullet$ (mm)	$d_6^\bullet$ (mm)	$d_7^\bullet$ (mm)	$d_8^\bullet$ (mm)	$d_9^\bullet$ (mm)
2D	79.0	14.3	4.9	0.8	9.4	19.7	23.8	9.7	7.8	4.1	5.6
V	79.0	10.9	8.9	1.7	15.0	-	-	-	-	-	-
Nabla	-	-	3.3	4.3	13.5	2.9	19.6	3.5	2.0	22.8	-

The final population became highly efficient while satisfying the torque constraints. There was a strong correlation between T_3^{pred} and T_4^{pred}. The maximum torque T_3^{req} at low speed was the most effective constraint in this setup. Severe torque constraints converge the population to a shape that can reduce copper loss. Thus, especially for 2D and V, the final population tended to have higher efficiency η_1^{pred} than η_2^{pred}, where copper loss is dominant rather than iron loss. On the other hand, for Nabla, the average torque tends to be the highest among the three topologies. Therefore, the effect of the torque constraints on the final population is not significant; the final population formed a broad Pareto front to η_1^{pred} and η_2^{pred}. A comparison of the characteristics of the three topologies indicates that efficiency η_1^{pred} at low speed tends to be highest for Nabla, where torque is easily obtained, and that efficiency η_2^{pred} at high speed tends to be highest for 2D, which has a high reluctance torque ratio.

FEA of the best shape was performed to verify the validity of the optimization results. Fig. 10 shows the selected best rotor shapes and Table 2 shows their geometric parameters. The best shapes are indicated by blue squares in Fig. 8. Fig. 11 shows the total losses under WLTC, where losses during regeneration were not considered. The optimized models achieved a loss reduction of 1.9% for 2D, 10.1% for V, and 5.6% for Nabla compared with the initial geometries. This is thanks to the accelerated computation time for individual evaluation, which allows many shape alternatives to be considered.

Table 3 shows the optimization computation time for each topology, where the results were measured for 10 optimization runs on a computer with an Intel Core™ i7-9700K CPU, 32.0 GB of RAM, and an NVIDIA GeForce RTX 3090 SUPER (24 GB) GPU. The optimization calculation took only about 2 hours even for 2D, which has 11 dimensions and the largest number of geometric parameters among three topologies. Thus, our proposed method can complete efficiency optimization with many parameters, which is
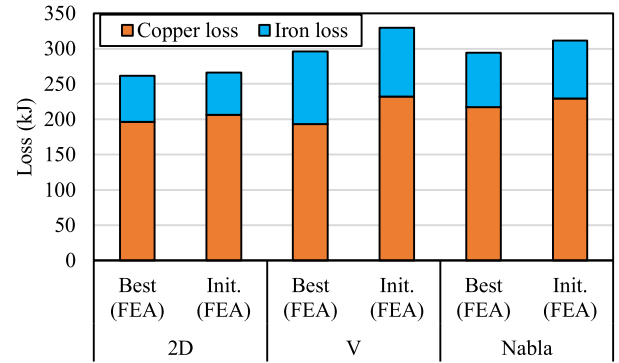

FIGURE 11. Loss under WLTC, where losses during regeneration were not considered.

TABLE 3. Computation time for optimization.

	2D	V	Nabla
Computation time (mean ± std)	123.58 ± 0.13 minutes	55.25 ± 0.84 minutes	89.30 ± 0.23 minutes

prohibitively time-consuming using FEA, in a relatively short time. Furthermore, once trained, the prediction model can be used repeatedly, allowing optimization under different conditions to be performed repeatedly in a short time.

V. CONCLUSION

This paper proposed a machine-learning-based surrogate model for predicting iron loss characteristics. This model allowed us to obtain the current conditions for the operating points with copper loss minimization control and to perform efficiency maximization design in a short time. The effectiveness of the proposed method was validated by IPMSMs with three rotor topologies.

The surrogate model takes the current and speed conditions as inputs in addition to geometry information. It achieved highly accurate iron loss prediction under a wide range of speed and torque conditions. The computation time of the surrogate model was less than 0.2% that of FEA. Therefore, the surrogate model allows an efficiency maximization design with multi-dimensional geometrical parameters to be performed, which is prohibitively time-consuming using FEA. The best shapes obtained in the optimization reduced the losses under the WLTC by 1.9%, 10.1%, and 5.6% compared with that for the initial motors for 2D, V, and Nabla, respectively. The computational time required for the optimization was 123.58 minutes even with 11-dimensional geometric parameters, indicating that the design can be completed in a short time. Furthermore, once trained, the prediction model can be used repeatedly, allowing optimization under different conditions to be performed repeatedly in a short time.

In future work, the driving characteristics of the optimized models will be verified using prototypes.

The dataset described in Section III-C is available at IEEE DataPort [16] and the Python implementation of the iron loss prediction model is available at GitHub [17].

REFERENCES

[1] T.-A. Huynh, P.-H. Chen, and M.-F. Hsieh, "Analysis and comparison of operational characteristics of electric vehicle traction units combining two different types of motors," *IEEE Trans. Veh. Technol.*, vol. 71, no. 6, pp. 5727–5742, Jun. 2022.

[2] A. Brosch, O. Wallscheid, and J. Bocker, "Torque and inductances estimation for finite model predictive control of highly utilized permanent magnet synchronous motors," *IEEE Trans. Ind. Informat.*, vol. 17, no. 12, pp. 8080–8091, Dec. 2021.

[3] Y. Shimizu, S. Morimoto, M. Sanada, and Y. Inoue, "Influence of permanent magnet properties and arrangement on performance of IPMSMs for automotive applications," *IEEJ J. Ind. Appl.*, vol. 6, no. 6, pp. 401–408, 2017.

[4] S. S. R. Bonthu, M. T. B. Tarek, and S. Choi, "Optimal torque ripple reduction technique for outer rotor permanent magnet synchronous reluctance motors," *IEEE Trans. Energy Convers.*, vol. 33, no. 3, pp. 1184–1192, Sep. 2018.

[5] M. Farhadian, M. Moallem, and B. Fahimi, "Multimodal optimization algorithm for torque ripple reduction in synchronous reluctance motors," *IEEE Access*, vol. 10, pp. 26628–26636, 2022.

[6] J.-C. Son, J.-Y. Kim, J.-W. Choi, D.-K. Lim, and H.-K. Yeo, "Performance enhancement of the IPMSM for HEV applications using grain-oriented electrical steel and design optimization," *IEEE Access*, vol. 10, pp. 46599–46607, 2022.

[7] Z. Shi, X. Sun, Y. Cai, and Z. Yang, "Robust design optimization of a five-phase PM hub motor for fault-tolerant operation based on Taguchi method," *IEEE Trans. Energy Convers.*, vol. 35, no. 4, pp. 2036–2044, Dec. 2020.

[8] M. S. Islam, M. Chowdhury, A. Shrestha, M. Islam, and I. Husain, "Multiload point optimization of interior permanent magnet synchronous machines for high-performance variable-speed drives," *IEEE Trans. Ind. Appl.*, vol. 57, no. 1, pp. 427–436, Jan. 2021.

[9] S. Zheng, X. Zhu, L. Xu, Z. Xiang, L. Quan, and B. Yu, "Multi-objective optimization design of a multi-permanent-magnet motor considering magnet characteristic variation effects," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 3428–3438, Apr. 2022.

[10] X. Sun, Z. Shi, G. Lei, Y. Guo, and J. Zhu, "Multi-objective design optimization of an IPMSM based on multilevel strategy," *IEEE Trans. Ind. Electron.*, vol. 68, no. 1, pp. 139–148, Jan. 2021.

[11] X. Sun, Z. Shi, Y. Cai, G. Lei, Y. Guo, and J. Zhu, "Driving-cycle-oriented design optimization of a permanent magnet hub motor drive system for a four-wheel-drive electric vehicle," *IEEE Trans. Transport. Electrific.*, vol. 6, no. 3, pp. 1115–1125, Sep. 2020.

[12] X. Sun, N. Xu, and M. Yao, "Sequential subspace optimization design of a dual three-phase permanent magnet synchronous hub motor based on NSGA III," *IEEE Trans. Transport. Electrific.*, early access, Jul. 13, 2022, doi: 10.1109/TTE.2022.3190536.

[13] H. Dhulipati, E. Ghosh, S. Mukundan, P. Korta, J. Tjong, and N. C. Kar, "Advanced design optimization technique for torque profile improvement in six-phase PMSM using supervised machine learning for direct-drive EV," *IEEE Trans. Energy Convers.*, vol. 34, no. 4, pp. 2041–2051, Dec. 2019.

[14] J. Hao, S. Suo, Y. Yang, Y. Wang, W. Wang, and X. Chen, "Optimization of torque ripples in an interior permanent magnet synchronous motor based on the orthogonal experimental method and MIGA and RBF neural networks," *IEEE Access*, vol. 8, pp. 27202–27209, 2020.

[15] Z. Pan and S. Fang, "Torque performance improvement of permanent magnet arc motor based on two-step strategy," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7523–7534, Nov. 2021.

[16] Y. Shimizu. (Aug. 19, 2022). *Dataset for Iron Losses of IPMSMs*. IEEE Dataport. Accessed: Aug. 19, 2022. [Online]. Available: https://ieee-dataport.org/documents/dataset-iron-losses-ipmsms

[17] Y. Shimizu. *Ironloss_Prediction_of_IPMSM_by_XGBoost*. Github. Accessed: Aug. 19, 2022. [Online]. Available: https://github.com/yshimizu12/ironloss_prediction_of_IPMSM_by_XGBoost

[18] Y. Shimizu, S. Morimoto, M. Sanada, and Y. Inoue, "Using machine learning to reduce design time for permanent magnet volume minimization in IPMSMs for automotive applications," *IEEJ J. Ind. Appl.*, vol. 10, no. 5, pp. 554–563, 2021.

[19] S. Suzuki, S. Morimoto, M. Sanada, and Y. Inoue, "Performance comparison of IPMSMs using a low iron loss material for automotive application," in *Proc. 19th Int. Conf. Elect. Mach. Syst. (ICEMS)*, Nov. 2016, pp. 1–6.

[20] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 785–794.

[21] C. P. Steinmetz, "On the law of hysteresis," *Proc. IEEE*, vol. 72, no. 2, pp. 197–221, Feb. 1984.

[22] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[23] J. Blank and K. Deb, "Pymoo: Multi-objective optimization in Python," *IEEE Access*, vol. 8, pp. 89497–89509, 2020.

YUKI SHIMIZU (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka Prefecture University, Sakai, Japan, in 2016, 2018, and 2022, respectively. In 2018, he joined with Toyota Motor Corporation, Aichi, Japan. Since 2022, he has been with the Graduate School of Science and Engineering, Ritsumeikan University, where he is currently an Assistant Professor. His research interests include design and control of permanent magnet synchronous motors using machine learning and deep learning. He is a member of the Japan Institute of Power Electronics and Society of Automotive Engineers of Japan.

• • •